Hamilton-Jacobi-Bellman (HJB) Equation (Justin Ting)

In these notes, I will be using a lot of approximations that you should be comfortable with.

① $f(x_0+dx) \approx f(x_0) + \partial_x f\, dx$ (Euler Approximation for small dx) ($\partial_x f$ is partial derivative with respect to x)

② $\int_0^{dx} f(\tilde{x})\, d\tilde{x} \approx f(0)\, dx$ (Leibniz Rule for small dx)

To understand the HJB equation, we need to understand what a Value Function is, and why it's important. A Value Function is defined as such:

③ $V(x_0, t_0) := \min_{u[t_0,t_f]} \left\{ \int_{t_0}^{t_f} \ell(x(t), u(t))\, dt + \varphi(x(t_f)) \right\}$

ignore

- $x \in \mathbb{R}^n$ is the state, and $x_0 = x(t_0)$ is the initial state.
- $u \in \mathbb{R}^m$ is the control, and $u[t_0,t_f]$ is a function defined on the interval $[t_0,t_f]$, so
$$u[t_0,t_f] := \{ u(t) \mid t \in [t_0,t_f] \}$$
- t_0 and t_f are initial and final times. Note that t_f is fixed, while t_0 is variable.
- $\ell(\cdot,\cdot)$ is a running cost. Integrating over $\ell(\cdot,\cdot)$ gives us a cost over a trajectory.
- $\varphi(x(t_f))$ is a terminal cost. Later we will see what happens when $\varphi(\cdot) = 0$.

④ $\dot{x} = f(x,u)$ is the time derivative of x.

Oftentimes we know how to penalize each point in time with $\ell(x(t), u(t))$, but trying to minimize the integral $\int_{t_0}^{t_f} \ell(x(t), u(t))\, dt$ requires a search over trajectories $u[t_0,t_f]$*. However, searching over trajectories is difficult. If possible, we should try to simplify so that we can evaluate a favorable $u(t)$ at only one point in time, using an evaluation of favorable states x.

Complex ☹		Simple ☺
Favorable Trajectories $u[t_0,t_f]$ $x[t_0,t_f]$	⟹	Favorable state x Favorable control $u(t)$ for one point in time

* There are ways to do this outside the scope of this lecture.

To evaluate favorable states x, we'd like a simple map $x \mapsto \mathbb{R}$ that weights the states, with higher weights being unfavorable. You can think of this like an energy function.

This is exactly what our Value Function ③ is doing, with a small twist. It is a function of both state and time, not just state. Therefore, the Value Function is a constant landscape function with respect to a fixed interval $[t_0, t_f]$.

The Value function reveals an interesting property if we split the integral by a small dt.

$$⑤ \quad \min_{u[t_0,t_f]} \left\{ \int_{t_0}^{t_0+dt} \mathcal{L}(x(t),u(t))\,dt + \int_{t_0+dt}^{t_f} \mathcal{L}(x(t),u(t))\,dt + \varphi(x(t_f)) \right\}$$

$$\overset{\text{Eqn } ②}{\approx} \min_{u[t_0,t_f]} \left\{ \mathcal{L}(x_0,u(t_0))\,dt + \int_{t_0+dt}^{t_f} \mathcal{L}(x(t),u(t))\,dt + \varphi(x(t_f)) \right\}$$

First, we define $u_0 := u(t_0)$. Notice the expression on the left $\mathcal{L}(x_0,u_0)$ only depends on u_0, so the expression can be rewritten as,

$$\min_{u_0} \left\{ \mathcal{L}(x_0,u_0)\,dt \right\} + \underbrace{\min_{u[t_0,t_f]} \left\{ \int_{t_0+dt}^{t_f} \mathcal{L}(x(t),u(t))\,dt + \varphi(x(t_f)) \right\}}_{\min_{u_0} \left\{ V(x(t_0+dt), t_0+dt) \right\} \quad ⑥}$$

The right expression is also a Value Function! It's identical to the original ③, except incremented forward by small dt. The integral is now only dependent on the interval $[t_0+dt, t_f] \to (t_0, t_f]$, but it is implicitly dependent on u_0 through the initial condition $x(t_0+dt) \approx x_0 + f(x_0,u_0)\,dt$. Therefore, we only need to minimize ⑥ over u_0, and the minimization over $u[t_0,t_f]$ happens through the definition of $V(x(t_0+dt), t_0+dt)$. Now we have,

$$V(x_0,t_0) = \min_{u_0} \left\{ \mathcal{L}(x_0,u_0)\,dt + V(x(t_0+dt), t_0+dt) \right\}$$

By the Euler Approx. ①, $V(x(t_0+dt), t_0+dt) = V(x(t_0) + f(x_0,u_0)\,dt, t_0+dt) = V(x_0,t_0) + \partial_x V^T(x_0,u_0) f(x_0,u_0)\,dt + \partial_t V(x_0,u_0)\,dt$

($\partial_x V$ is a gradient)

Using this approximation, dropping the (x_0,u_0) dependencies in V for readability,

$$V(x_0,t_0) = \min_{u_0}\left\{ \mathcal{L}(x_0,u_0)dt + \partial_x V^T f(x_0,u_0)dt + \partial_t V dt \right\}$$ [redacted] $+ V(x_0,t_0)$

↳ not dependent on u_0, so move outside min-brackets

Cancel $V(x_0,t_0)$ on both sides

$$dt\left[\partial_t V + \min_{u_0}\left\{ \mathcal{L}(x_0,u_0) + \partial_x V^T f(x_0,u_0)\right\}\right] = 0$$

↳ must be 0,

$$\Rightarrow \quad -\partial_t V = \min_{u_0}\left\{ \mathcal{L}(x_0,u_0) + \partial_x V^T f(x_0,u_0)\right\} \quad \leftarrow \text{HJB Equation.}$$

Now we have an equation to relate x_0 and u_0, instead of $u[t_0,t_f]$, $x[t_0,t_f]$.

Note on terminal cost $\varphi(x(t_f))$.

$t_0 = t_f \rightarrow \int_{t_f}^{t_f} \mathcal{L} = 0$

Note that $V(x_0,t_f) = \varphi(x(t_f))$, since the integral collapses. So the shape of $\varphi(x)$ has an impact on the system's behavior near t_f. This is important in cases like MPC where you expect your system to continue acting past t_f.

We can derive a "near-t_f" HJB with similar methods as before.

$$V(x_0, t_f - dt) = V(x_0,t_f) - \partial_t V\, dt = \min_{u[t_f-dt,\, t_f]}\left\{ \int_{t_f-dt}^{t_f} \mathcal{L}(x(t),u(t))dt + \varphi(x(t_f))\right\}$$

(using ①) (using ③)

$$= \min_{u[t_f-dt,\, t_f]}\left\{ \mathcal{L}(x_0,u_0)dt + \varphi(x_0 + f(x_0,u_0)dt)\right\} = \varphi(x_0) + \min_{u_0}\left\{ \mathcal{L}(x_0,u_0) + \partial_x \varphi^T f(x_0,u_0)\right\}dt$$

Since $V(x_0,t_f) = \varphi(x_0)$, cancelling on both sides gives you the HJB equation. For the $\min_{u_0}$ condition to be satisfied, we take the gradient of $\mathcal{L}(x_0,u_0) + \partial_x\varphi f(x_0,u_0)$ and set it to 0.

$$\partial_u \mathcal{L}(x_0,u_0) + \partial_x \varphi(x_0)^T \partial_u f(x_0,u_0) = 0$$

Here, we can see that the optimal value of u_0, which represents the value of $u(t)$ for any t near t_f, the terminal cost induces a significant offset and tradeoff. The term $\partial_u f(x_0,u_0)$ emphasizes the relevance of x_f implicitly. This is much easier to see in the discrete-time case.

~~In the discrete case, it is easy to see that without terminal cost, u_f = 0 [illegible]~~